

Amendments to the Specification:

Please replace the paragraph (or section) beginning at page 23, line 19, with the following redlined paragraph (or section):

Different mechanisms may be used by the game client system (and/or ad measurement routines 156) to measure ad exposure quantity and quality. Preferably, since a client game already needs to track position of the player relative to each object and the distance of the player from the object, these same techniques and/or measurements can be used. Typically, these mathematical and heuristical techniques are known in the art for determining what objects are to be displayed (or eliminated from display) when the player changes his/her game view. For example, these techniques are used to avoid painting objects on the output device that are outside the player's viewing range. Building C 330 in Figure 3A is an example of such an object within the depicted scenario. In the ad insertion scenario, these same techniques can be used to supply the quality and quantity of ad exposure. Example techniques include: use of a ~~frustrum~~-frustum model (*e.g.*, a view frustum), trigger maps, and hit testing based upon geometry divisions of the display screen. Alternatively, code can be added into each player instance to track needed movement information, which is then translated for its effect relative to the ad. Also, ads can be dynamically “wrapped” with code to measure their view of each player. One skilled in the art will recognize that techniques for measuring point of view and exposure in other 3D virtual worlds can be similarly extended to measuring the effectiveness of inserted content.

Please replace the paragraph (or section) beginning at page 24, line 19, with the following redlined paragraph (or section):

Figure 4 is a block diagram of a general purpose computer system for practicing embodiments of a client portion of the dynamic inserter. Specifically, the block diagram of Figure 4 shows a computer system that performs as a game client system. The computer system 400 includes a central processing unit (CPU) 402, a display 404, an audio output 406, a local storage device 410, as well as other input/output devices 408. The computer system 400 also includes a computer memory 420. Components of the dynamic inserter 421 typically reside in

the computer memory 420 and execute on the CPU 402. As described with reference to Figure 2, the portion of the dynamic inserter that runs on the game client system includes ad inserter routines 422, and may optionally include ad measurement routines 426 and temporary ad storage 424. Additionally, a video game 430 or some other executable program also resides in the computer memory 420 and executes on the CPU 402. Also, in some embodiments enhanced game objects, such as enhanced game objects 158 in Figure 1B, reside in the computer memory 420 and execute on CPU 402. Other programs 428 also may reside in the memory 420. One of the input/output devices 408 is the connection means, such as a modem or other hardware that allows the computer 400 to connect to the communication network. One skilled in the art will also recognize that exemplary dynamic inserters 421 can be implemented as one or more code modules, may be implemented in a shared environment with multiple displays 404 coupled to the same computer memory 420, or may be implemented in a distributed environment where one or more of the components reside in different locations. In addition, one skilled in the art will recognize that the components of an exemplary dynamic inserter 421 may be combined or that additional components may exist.